Adaptive Radiation of Alife Memes: Review of Eight Artificial Life
Book/Software Packages

Peter M. Todd
Department of Psychology
University of Denver
2155 S. Race Street
Denver, Colorado 80208
E-mail: ptodd@pstar.psy.du.edu

\fICyberlife\fP
ISBN 0-672-30491-0.

\fIWindows Hothouse: Creating Artificial Life with Visual C++\fP
by Mark Clarkson.
xiv + 288 pages, 3.5" IBM disk of Windows source code (requires Visual C++ to
compile and run), $34.94 paperback.

\fIIt's Alive!: The New Breed of Living Computer Programs,\fP
by Frederick B. Cohen.
xii + 153 pages, 3.5" Macintosh disk of C and Unix software, $39.95 paperback.

\fIDeveloping Dinosaurs and Ancient Worlds,\fP
by Rick Leinecker.
xx + 254 pages, 3.5" IBM disk of C source code and compiled software, $29.99
paperback.

\fIArtificial Life Playhouse: Evolution at Your Fingertips,\fP
by Stephen Prata.
viii + 179 pages, 3.5" IBM disk of compiled software, $23.95 paperback.

\fIArtificial Life Lab,\fP
by Rudy Rucker.
xvi + 283 pages, 3.5" IBM disk of compiled software and 3D glasses, $34.95
One of the hallmarks of evolving, adaptively behaving life is its ability to radiate into new environmental niches and fill them with seething activity. Artificial life is no exception. A variety of species of artificial creatures have evolved over the computer-cycle millennia in the primordial ooze of Unix, spreading slowly but steadily from habitat to habitat. But in recent technological time, a vast new inland sea has formed, much broader (though shallower?) than the area covered by the Unix waters. It was only a matter of time before this PC sea was colonized by intrepid artificial life forms venturing out from their original environment. This sea is now filled with the rapidly evolving and radiating progeny of those first invaders. Some of these species can form the basis for useful research, some can be used for educational purposes, and still others might provide entertainment and diversion, so learning about the available compu-diversity has potential rewards. But anyone intending to explore this increasingly tangled bank (to mix metaphors slightly) would do well to find a guide. This review will summarize some of the recent evolution of PC-based artificial life, and identify and critique a few of its most prolific strains.

The relatively slow spread of artificial life in the Unix environment can be attributed at least in part to the limiting vectors through which it reproduced. Alife organisms, for the most part, did not spawn live copies and disperse whole; rather, they laid memes that had to be carried through some appropriate medium--text, video, or conference presentation--into a sophisticated host willing and able to create a new instance of the original organism. That is, often alife researchers would "simply" code up a new version of a simulation they read about, rather than hassling with understanding (and converting) someone else’s code. In this way, of course, significant variation could be introduced into the alife-system evolutionary process, but the bottleneck of available host researchers limited system population sizes and the speed at which their evolution could proceed.

In the PC environment, though, a new "evolutionary watershed" innovation (as coined by Richard Dawkins) has appeared. Rather than spreading via memes from host to host, alife books have sprouted phenotypes: actual working (living?) copies of artificial life simulation systems provided on disks or CD-ROM. These alife simulation systems are widely available from several publisher/breeders, leading to rapid dispersal and large population sizes. But of course every book/software individual within a given species is a clone, an exact copy of all the others of its
kind, so it is certainly not immediately clear whether or not the greater numbers of alife systems in the PC environment will make for a greater rate of their evolution than in the Unix environment.

Still, the several species that have recently emerged are all more or less distinct and uniquely adapted to their particular niche and audience. (After all, a minimal sort of host must still be recruited at least to install the alife software onto a new patch of the global PC environment.) One of the most unique and sophisticated of the current offerings is Rucker’s \textit{Artificial Life Lab}. Rather than providing and describing a variety of individual, small-scale alife simulators, as most of the other texts do, Rucker presents his ambitious Boppers program that incorporates many important alife concepts at once. Boppers are artificial creatures that can be controlled by a combination of finite-state machines and Reynolds-inspired "boid" flocking and movement algorithms. The three possible simultaneous colonies of Boppers can evolve through mutation, crossover, and gene sharing between colonies, in response to fitness functions based on environmental food/poison, predator/prey species interactions, and other factors including movement and behavioral entropy. The environment can also incorporate patches that grow using a variety of cellular automata (CA) rules, to show how Boppers-based behavior and CA-based "physics" can interact. Rucker writes lucidly and insightfully about all of these topics, providing a strong foundation in the concepts underlying alife and adaptive behavior. The Boppers world is fascinating to watch, as a rich range of species and interactions is possible; but because of all the various components involved, it is a bit difficult to learn about any one of them (say, CAs) in isolation. This package is perhaps best used first to read and learn about the tools used in alife, and then to see the variety of behavior that can emerge when the tools are combined, rather than to explore any one tool in particular.

Cohen's book is also distinctive, focusing on the world of computer viruses, or living programs as he renames them, rather than on the broader range of alife topics. He describes his own work on computer viruses over the years, as well as other systems, including Tom Ray’s Tierra and Kee Dewdney’s Core War, in which programs live and compete. But there is rather little detail in the book, and much of what is included is confusing (his mathematical definition of viruses and his code fragments are rather cryptic without further explanation). A good third of the book is loose speculation about life, robots, and transporter beams (though some of it is interesting, e.g., evolution as noisy communication). The accompanying disk, containing programs for two versions of Core War, linear and spatial CAs, and a few Unix-based viruses, seems to have been mostly an afterthought, as there is virtually no mention of it in the text after a couple of brief paragraphs in the introduction. Cohen's other book, \textit{A Short Course on Computer Viruses}, is perhaps a better choice on this topic.

As with most evolving systems, successful components--here, popular alife programs--that appeared early on are preserved in later incarnations: all of these packages, in particular, contain some form of Conway's Game of Life. In some instances, these components have undergone developmental doubling and other forms of mutation. Prata's book, for example, contains two versions of Life, one for DOS and one for Windows. The latter version is one of the nicest of the bunch, in that it allows exploration of different types of rules (also an aspect of the Life program in Leinecker's book, which additionally includes a form of learning to adjust the rules used). Prata includes a variety of other simple simulations, including other CAs, a cumulative selection program similar to Dawkins's Blind Watchmaker biomorph evolver, a system modeled after Michael
Palmiter’s evolving “Bugs,” and two ant-colony simulations whose inhabitants evolve using a genetic algorithm. Each program is thoroughly described and used to explain a particular set of ideas from artificial life research. As the book’s title hints, the level overall is fairly introductory, and some of the software is a bit more buggy than its designers may have intended, but all told this is a good educational package.

Evolution is a noisy, stochastic process, and good genes (or ideas) are no guarantee of continued viability or success. Thro’s book, for instance, despite providing perhaps the best overview of actual research in artificial life and adaptive behavior, is currently out of print (though it is still widely available in bookstores and through the Media Magic catalog, which carries most of the titles reviewed here, at 1-800-882-8284). Thro gives a good introduction to the tools of these fields, including neural networks, genetic algorithms, classifier systems, and CAs. She then proceeds to describe research from Langton’s self-replicating loops, to Wilson’s animats and Beer’s cockroaches, to Ackley and Littman’s study of the interaction of evolution and learning, all in a palatable fashion. The disk comes with five CA programs, an early version of Rucker’s Boppers program dedicated to Turing-machine ants, and a rather involved environmental simulation of an aquatic ecosystem called BSIM. Some of the programs could be better described, and more discussion of the principles that they demonstrate would be helpful (along the lines of Prata), but the package is worth looking into.

Rather than narrowing in on a specific audience niche, some of the alife book/software species have become generalists aimed at a wide range of readers. Walnum’s book, for example, includes chapters on not just alife, but also computer viruses, artificial intelligence, robotics, and virtual reality. But generalists tend to trade off their broad range against a precise fit to any particular niche, and this package too suffers in that regard. The tone is a bit sensationalist (how shocking and disturbing \lIs\lP artificial life, really?), many of the figures are unnecessary, and some of the facts are off (for instance, it is certainly controversial at least whether or not crossover slows down evolution). In addition to a simple Life program, the Microants program (also included in Prata), and a virtual reality demo, the package includes more peculiar software such as several computer virus demonstrations and an expert system that gives fishing advice (to demonstrate AI techniques). One nice addition is a system that simulates Beer’s wall- and scent-following artificial cockroach, complete with neural activity tracking and the ability to add new neurons. But the text gives very little detail about how to use these features, and instead recommends reading Beer’s book for guidance, which is probably good advice. The book \lICyberlife\lP (the only one to come with a CD-ROM rather than a disk) is even more scattershot, aiming at all manner of currently hot cyber-topics. In addition to virtual reality, net-sex, automated music and art, and the digital workplace, there are chapters on intelligent agents, artificial life (with an emphasis on Lamarckian evolution), and computer viruses, of varying quality. Each chapter is of necessity limited in depth, and most fail to discuss the related software (which includes Life and the enigmatic BSIM for the alife chapter, along with other things such as morphing and virtual reality demos and Doom), but the book as a whole might be useful for a novice interested in what cool things computers in general can do.

Most of these alife book/software packages are evolutionary dead ends. That is, they only include the cloned compiled-software phenotype, and not the underlying source code genotype that would allow further modification and evolution of the system. Leinecker’s disk, though, does include C
source code for the simulations he covers, and the book lists portions of this code and describes it through the included comments. The simulations included are another version of Life, with the ability to create complex CA (and "learning-adjustable") rules, and a complex whole-world ecosystem simulator, Ecosim, which includes dynamically changing geological features, weather patterns, and flora and fauna. Ecosim emphasizes creating this "ancient world" with volcano-strewn tectonic plates and occasional ice-ages, and populating it with prehistoric plants and animals, such as cycads and dinosaurs. Building an environment up from its physical features to its biological ones is a nice idea, not dissimilar to Maxis's SimLife program. But Leinecker's text largely leaves the reader hanging after the world has been created and populated--there is little discussion of how to make it actually run over time, what to look for, or what to learn as the simulated millennia tick by. For instance, one can design dinosaurs that can apparently evolve under a variety of selection regimes, including convergent evolution, disruptive selection, and sexual selection, and these dinosaurs can be endowed with a variety of kinds of learning, including imprinting, conditioning, and insight. But nowhere is it made clear how either of these processes is actually put into effect in the world (let alone implemented in the code), nor how they will alter the course of life on this simulated planet. Nor is there much dynamic visual feedback from the simulation's several windows to get a handle on how the changes to a particular species might affect the overall balance of nature. While Leinecker has made efforts to create an educational system for students (there are science fair project suggestions, and other more game-oriented programs such as a dinosaur quiz, included in the package), the creators of SimLife need not worry just yet.

Finally, Clarkson's book and software provide probably the best introduction to the actual practice of artificial life and adaptive behavior research, by teaching the reader explicitly how to write a variety of types of simulations in Visual C++. It might help to know a little programming before diving into this book, but it is not absolutely necessary--the package is intended to get even beginning programmers putting interesting alife systems up on the screen as rapidly as possible. It is one thing to read about finite-state automata, and watch them churn away in a canned simulation; it is quite another, more educational, thing actually to build one in software and see what makes it tick. Because Clarkson wants to teach readers not only about alife, but also about object-oriented Windows programming, there is not room to cover quite as many alife topics in this book. But he still manages to cover a surprising variety, proceeding from (of course) the Game of Life, to genetic algorithms for evolving bug-eating frogs, to a (very) simple version of Braitenberg's Vehicles, to L-systems to grow plant morphologies, to Dawkins's interactive biomorph evolution, finally to an instantiation of Dewdney's Core War. Clarkson writes clearly and with humor (e.g., discussing whether nonreproducing mules and popes are actually alive), and he provides a variety of interesting ideas for directions to develop and explore further the programs he covers. For anyone interested in really learning how to create useful artificial life simulators, this would be a good book to turn to.

Provided, of course, that one wants to develop simulations in Visual C++ under Windows. This certainly may be a very appropriate choice from the standpoint of portability and perhaps even ease of development. But the existence of Clarkson's book, and the others described here, highlights the lack of such texts for the Unix environment. While the relatively narrow confines of the Unix ocean may not be able to support the diversity of alife book/software packages now filling the PC sea, there should be room for at least one or two successful individuals. A book that combined the
coverage of alife/adaptive behavior tools of Rucker, with the description of research projects of Thro, and the insights into programming of Clarkson, coupled with interesting, working software to demonstrate all of these things, would be a very useful package indeed. Given the ongoing convergent evolution of operating systems themselves, such a package need not even be OS-specific, and could incorporate developmental switches that let it "grow" into software phenotypes appropriate to any of a set of particular operating environments. By giving this "hopeful monster" conglomeration of artificial life and adaptive behavior memes the ability to invade a variety of environments, the ongoing evolution and exploration of these ideas could be ensured.¹